

コンピュータを用いたハイランド円線図の作図

古賀 亜彦

To Draw the Heyland Circle Diagram by Using a Computer

Tsuguhiko KOGA

The purpose of this paper is to draw the Heyland circle diagram by using a computer. This diagram shows us various kinds of characteristics of an induction motor. It is very important for us to get many kinds of characteristics of an induction motor without applying real load to it. This diagram is based on a simple equivalent circuit of an induction motor and named Heyland circle diagram. It is very difficult to draw the Heyland's circle diagram, but by observing this program displayed, we can understand and master how to draw the diagram. This program is very useful for studying of the electric machinery. Visual C++6.0 was used to write this program.

Key Words : Heyland Circle Diagram, Induction Motor, Electric Machinery, Visual C++6.0

1. 目的

誘導電動機の諸特性を表すものにハイランド円線図というものがある。この円線図は誘導電動機の簡易等価回路図に立脚したものであり簡便なためによく利用される。大型の誘導電動機の場合、大きな実負荷をかけることは困難な場合が多い。そのような時、ハイランド円線図を利用すれば任意の負荷に対するすべての特性がわかる。本校の電気機器実験においてもハイランド円線図のレポートを提出させている。人間の手で

ハイランド円線図を描くのは結構面倒であり、難しいことである。そこで、このハイランド円線図をコンピュータを用いて描き、コンピュータ上で任意の負荷に対する誘導電動機の特性をシミュレートしようとする事が今回の目的である。この研究は誘導電動機の特性をパソコン上でビジュアルに追跡できるので、誘導電動機の授業および実験において非常に有益であると考ええる。

2. コンピュータの機器構成および使用言語

パソコンは OS がウィンドウズで、処理言語は

マイクロソフト社の VC++ 6.0 を使用した。また、プログラムの大きさは 1024 ステップである。

3. 誘導電動機の基礎実験データ

対象となる誘導電動機は、6 極、60 Hz、3 相 Y 結線であり、ハイランド円線図の基礎となる 3 つの実験データは、

(1) 抵抗測定 : Y 結線一相の抵抗 $R_t = 4.6 \Omega$ 、

室温は $t = 17^\circ\text{C}$ 、

(2) 無負荷試験 : $V_n = 3,000 \text{ V}$ 、 $I_{00} = 5.88 \text{ A}$ 、
 $P_{00} = 2.258 \text{ kW}$ 、

(3) 拘束試験 : $V_s' = 710 \text{ V}$ 、 $I_s' = 19 \text{ A}$ 、
 $P_s' = 6.075 \text{ kW}$ であった。

4. 誘導電動機の計算値データ

上記の基礎実験結果から下記の計算結果を得る。

75°C に換算した一相の抵抗 $r_1 = (R_t/2) * (234.5 + 75) / (234.5 + t) = 2.83 \Omega$

無負荷電流の有効電流 $i_{01} = P_{00} / (\sqrt{3} * V_n) = 0.435 \text{ A}$

無負荷電流の無効電流 $i_{02} = \sqrt{(I_{00}^2 - i_{01}^2)} = 5.86\text{A}$

定格電圧に対する短絡電流 $I_s = I_s' \cdot (V_n/V_s') = 80.3\text{A}$

定格電圧に対する短絡電力 $P_s = P_s' \cdot (V_n/V_s')^2 = 108\text{kW}$

定格電圧に対する短絡電流の有効電流 $i_{s1} = P_s / (\sqrt{3} \cdot V_n) = 1.17\text{A}$

定格電圧に対する短絡電流の無効電流 $i_{s2} = \sqrt{(I_s^2 - i_{s1}^2)} = 80.29\text{A}$

4. 倍率の決定

作図の前に倍率を決めなければならない。線分 OB [$= i_b = i_{s2} + (i_{s1} \cdot i_{01}) / (i_{s2} \cdot i_{02})$] が一番長くなるので、それを MAX(=800)に対応させる。

即ち、 $\text{factor} = \text{MAX} / i_b = 9.60 [\text{dots} / \text{A}]$ が、倍率となる。言い換えると、1A は 9.6 ドットに相当するということである。以後すべての計算において factor 倍される。

5. ハイランド円線図の基本図形の作図

下記の作業手順に従ってハイランド円線図の基本図形 (図 1) を描く。

- (1) $ON = i_{01} - j i_{02} = 0.435 - j 5.86\text{A}$ より点 N が決まり、
- (2) $OS = i_{s1} - j i_{s2} = 1.17 - j 80.29$ より点 S が決まる。
- (3) 2 点 N、S を通り線分 NB を直径とする半円を描く。
- (4) 点 S から線分 NB に下ろした垂線の足を U とし、
- (5) $TU = (\sqrt{3}) \cdot I_s^2 \cdot r_1 / V_n$ から計算される点 T を決定する。TU は起動時の全 1 次銅損にあたる。
- (6) 線分 SN の延長線と線分 OX との交点を D とする。
- (7) 点 S から線分 OY に垂線 SS' を引き、
- (8) 点 D から線分 SS' に垂線 DF を引く。点 F は定点となる。
- (9) 線分 SF は効率を表し、これを百等分する。
- (10) 次に、「点 N から垂直に引いた直線」と「点 S から線分 TN に平行な直線」との交点を G とする。
- (11) 点 G は定点となり、線分 GS は滑りを表し、これを百等分する。

6. 作図のために用意したサブルーチン

作図のために、下記のプログラムを作成した。

- (1) **void arc(CDC *pDC, double cx, double cy, double r, double a1, double a2)**

機能：円弧を描く

引数：cx, cy：円の中心座標、r：半径、a1：開始角度[°]、a2：終止角度[°]

- (2) **void circle(CDC *pDC, double cx, double cy, double r)**

機能：円を描く

引数：cx, cy：円の中心座標、r：半径

- (3) **void line(CDC *pDC, double x1, double y1, double x2, double y2)**

{

pDC->MoveTo(X0+(int)(x1*factor), Y0-(int)(y1*factor));

pDC->LineTo(X0+(int)(x2*factor), Y0-(int)(y2*factor));

}

機能：論理座標(x1,y1)、(x2,y2)に倍率 factor を掛けてドット座標に変換して、かつ画面左下(X0, Y0)を図形の原点となるように校正して線分を引く。

引数：(x1, y1)：出発点、(x2, y2)：終点

7. 作図に必要な座標点を計算するためのサブルーチン

作図に必要な座標点を計算するために「直線と直線の関係を計算するプログラム」、「直線と円の関係を計算するプログラム」、「円と円の関係を計算するプログラム」を 10 本作成した。

(1) **int tanc1(double xc, double yc, double r, double x1, double y1,
double *a, double *b, double *c, double *xt, double *yt, short kp)**

機能：与えられた円に対して任意の点より引いた接線の式と接点の座標を求める

引数：xc, yc, r：与えられた円の式

x1, y1：与えられる任意の点の座標

xt, yt：求められる直線と円の交点の座標

a, b, c：求められる直線の式

kp = 1：x座標が大きいほうを選択

x座標が等しいときは、Y座標が大きいほうを選択

kp = 2：x座標が小さいほうを選択

x座標が等しいときは、Y座標が小さいほうを選択

kp = 3：y座標が大きいほうを選択

y座標が等しいときは、X座標が大きいほうを選択

kp = 4：y座標が小さいほうを選択

y座標が等しいときは、X座標が小さいほうを選択

戻り値 正常終了の時 0

異常終了の時 -1

(2) **int cross3(double a, double b, double c, double xc, double yc, double r, double *xx, double *yy, short kp)**

機能：直線と円の交点を求める

引数：a, b, c：与えられた直線の式 ($aX + bY + c = 0$)

xc, yc, r：与えられた円の式

xx, yy：求められる直線と円の交点の座標

kp = 1：x座標が大きいほうを選択

x座標が等しいときは、Y座標が大きいほうを選択

kp = 2：x座標が小さいほうを選択

x座標が等しいときは、Y座標が小さいほうを選択

kp = 3：y座標が大きいほうを選択

y座標が等しいときは、X座標が大きいほうを選択

kp = 4：y座標が小さいほうを選択

y座標が等しいときは、X座標が小さいほうを選択

戻り値 正常終了の時 0

異常終了の時 -1

(3) **int cross4(double xc1, double yc1, double r1, double xc2, double yc2, double r2,
double *xx, double *yy, short kp)**

機能：与えられた2つの円の交点を求める

引数：kp = 1：x座標が大きいほうを選択

x座標が等しいときは、Y座標が大きいほうを選択

kp = 2：x座標が小さいほうを選択

x座標が等しいときは、Y座標が小さいほうを選択

kp = 3：y座標が大きいほうを選択

y座標が等しいときは、X座標が大きいほうを選択

kp = 4：y座標が小さいほうを選択

y座標が等しいときは、X座標が小さいほうを選択

xc1, yc1, r1：与えられた円1の式

xc2, yc2, r2 : 与えられた円 2 の式

xx, yy : 2 円の交点の座標

戻り値 正常終了の時 0

 異常終了の時 -1

(4) **void det2(double x1, double y1, double x2, double y2, double *x, double *y, short kp)**

機能 : 解が 2 個あるとき、条件(kp)を満足するほうを選択する

引数 : kp = 1 : x 座標が大きいほうを選択

 x 座標が等しいときは、Y 座標が大きいほうを選択

kp = 2 : x 座標が小さいほうを選択

 x 座標が等しいときは、Y 座標が小さいほうを選択

kp = 3 : y 座標が大きいほうを選択

 y 座標が等しいときは、Y 座標が大きいほうを選択

kp = 4 : y 座標が小さいほうを選択

 y 座標が等しいときは、Y 座標が小さいほうを選択

(5) **void lvert2(double x1, double y1, double x2, double y2, double *a, double *b, double *c)**

機能 : 与えられた線分の垂直二等分線の直線の式を求める

引数 : (x1, y1)、(x2, y2) : 与えられた 2 点の座標

 a, b, c : 求められる直線の式

(6) **void midle(double x1, double y1, double x2, double y2, double *xx, double *yy)**

機能 : 2 点の midpoint を求める

引数 : (x1, y1)、(x2, y2) : 与えられた 2 点の座標

 xx, yy : 求められる midpoint の座標

(7) **void lpoin(double x1, double y1, double x2, double y2, double *a, double *b, double *c)**

機能 : 2 点を通る直線の式を求める

引数 : (x1, y1)、(x2, y2) : 与えられた 2 点の座標

 a, b, c : 求められる直線の式

(8) **void lvert(double x1, double y1, double a1, double b1, double c1, double *a, double *b, double *c, short k)**

機能 : 1 点を通り 与えられた直線に平行な直線、または垂直な直線の式を求める

引数 : x1, y1 : 与えられた点

 a1, b1, c1 : 与えられた直線

 a, b, c : 求める直線の式

 k = 1 : 平行な直線

 k = 2 : 垂直な直線

(9) **void lfoot(double x1, double y1, double a1, double b1, double c1, double *xx, double *yy)**

機能 : ある点から 与えられた直線に下ろした垂線の足の座標を求める

引数 : x1, y1 : 与えられた点

 a1, b1, c1 : 与えられた直線

 xx, yy : 求める垂線の足の座標

(10) **void cross1(double a1, double b1, double c1, double a2, double b2, double c2, double *xx, double *yy)**

機能：2直線の交点を求める

引数：a1, b1, c1：与えられた直線1

a2, b2, c2：与えられた直線2

xx, yy：求める交点の座標

8. 誘導電動機のハイランド円線図の基本図

誘導電動機の3つの実験から描けるハイランド円線図の基本図を図1に示す。この段階で効率の目盛り、滑りの目盛りが作図できる。

9. 任意の出力に対するハイランド円線図

(1) 出力75 kWに対するハイランド円線図を図2に示す。この時の諸特性は次のとおりである。

出力=75 kW

力率=86.78%

効率=91.52%

滑り=3.37%

トルク=77.61 kW

(2) 出力150 kWに対するハイランド円線図を図3に示す。この時の諸特性は次のとおりである。

出力=150 kW

力率=77.76%

効率=80.38%

滑り=11.38%

トルク=169.14 kW

10. 特殊な場合の誘導電動機のハイランド円線図

次に、特殊な条件の場合の諸特性を表すハイランド円線図を図4に示す。

図中のP1~P5はそれぞれ次の場合の運転状態を表している。

P1：最大効率を発揮する運転状態で、この時の効率は92.25%である。

P2：最大力率を発揮する運転状態で、この時の力率は87.33%である。

P3：最大出力151.92 kWを表す運転状態である。

P4：最大トルクを発揮する運転状態で、この時のトルクは177.48 kWである。

P5：最大1次入力を発揮する運転状態で、この時の1次入力は203.57 kWである。

11. 考察

ハイランド円線図の作図は学生にとってはかなり難しい作業であり、またあまり面白くないものである。今回の、パソコンによる画面を学生に見せると興味を示す。自分で計算した結果と作図が一致すると大変喜ぶ。また、答えが合ったことで安心できる。出力をいろいろ変化させて効率や力率等の変化を調べると、ある特殊な状態における最適な運転条件はどの時であるかが判定できる。誘導電動機のシミュレーションが可能となる。一般的に出力が大きくなると滑りが大きくなることは分かっている。即ち、出力が小さいほど滑りは小さい、しかしその他の状態が悪化する。効率および力率が最大になるのはそれぞれ最適な運転時である。また、1次入力、2次入力、2次出力が最大になるのもそれぞれ最適な運転をしたときである。我々が、直接監視できるのは出力であるから、いくら出力の時、それぞれの値が最適になるかが分かっているれば目的の最適運転が可能なのである。図4がそれを表している。

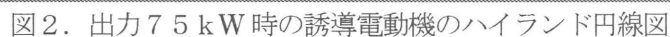
なお、作画の精度についてであるが、「計算上出した結果」と「作図からの読取値」はほとんど区別がつかないくらいに正確であった。したがって、誘導電動機の演習および実験を補強するものとして十分使用に耐えうるものであることが確認できた。

今回の教育用プログラムは、学生の誘導電動機の学習にとって非常に有意義であり、このハイランド円線図のプログラムを電気機器用教育用プログラムとして、今後の誘導電動機の学習に使っていきたい。

ウィンドウズ、VC++6.0は、マイクロソフト社の登録商標である。

参考文献

後藤文雄：電機概論、丸善株式会社



コンピュータを用いたハイランド円線図の作図

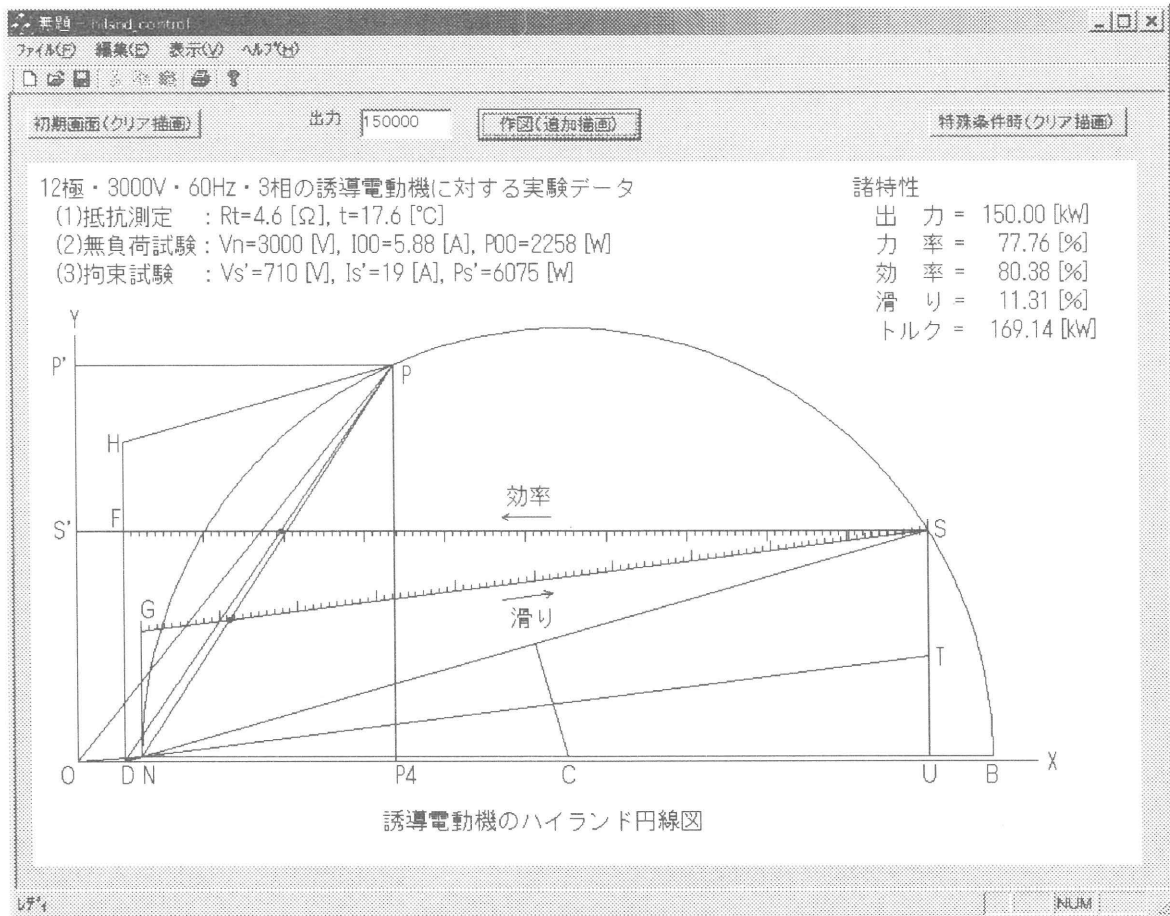


図3. 出力150kW時の誘導電動機のハイランド円線図

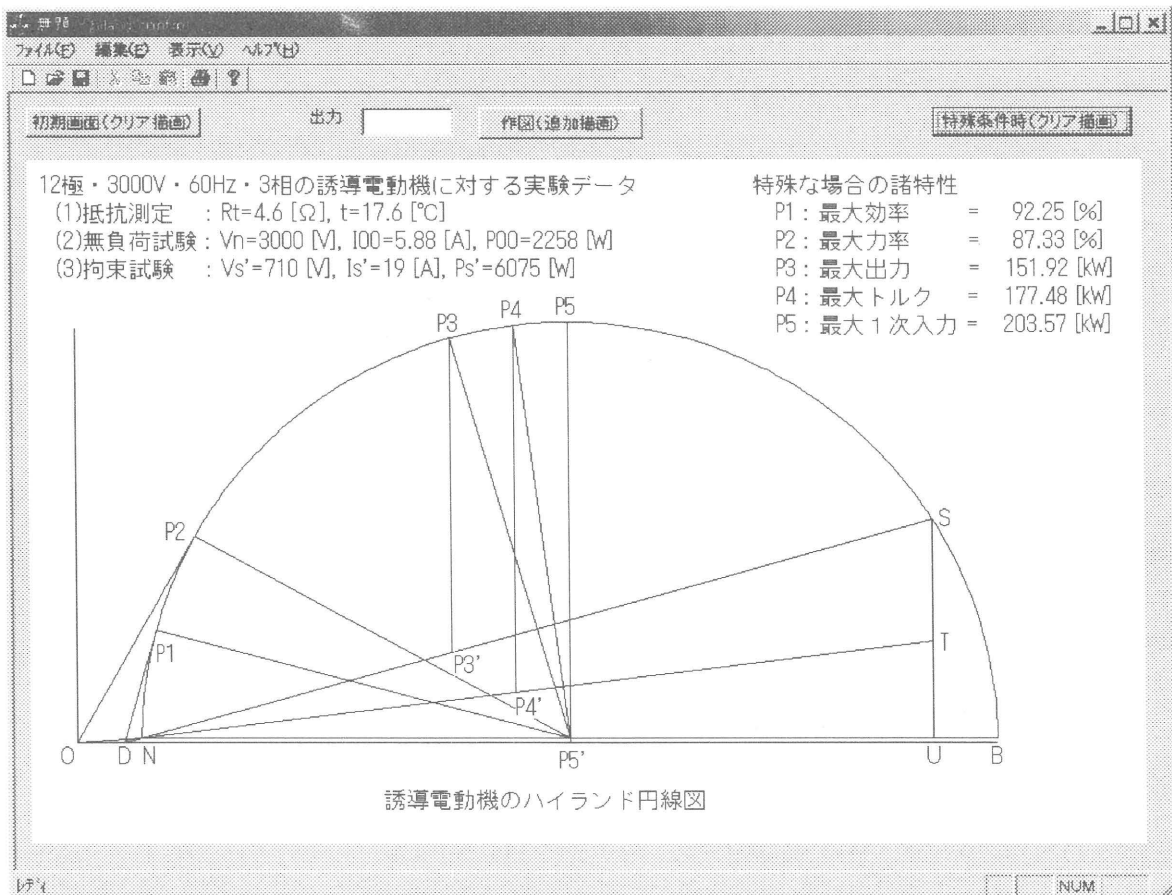


図4. 特殊な条件の時の誘導電動機のハイランド円線図